

# Estilos CSS: dónde y cómo crearlos

## Dónde crearlos

1. **Al vuelo:** todas las etiquetas HTML admiten el atributo **style=""**, introduciendo los atributos css y sus valores dentro de las comillas para definir los estilos de esa etiqueta. Ejemplo:

```
<h1 style="font-size:18px; color:#f00;">texto</h1>
```

2. **En el <head>:** en la cabecera del documento podemos abrir una región **<style type="text/css"></style>** para definir estilos css para ese documento. Ejemplo:

```
<style type="text/css">
<!--

-->
</style>
```

3. **En un documento css externo:** creamos un documento con extensión **.css** que tenga toda la definición de estilos, y enlazamos los documentos **.html** con una llamada en el **<head>** del documento tipo **<link href="[ruta]" rel="stylesheet" type="text/css" />**. Ejemplo:

```
<link href="css/estilos.css" rel="stylesheet"
type="text/css" />
```

También podemos utilizar la regla **@import url([ruta]);** dentro de una etiqueta **<style>**. Ejemplo:

```
<style type="text/css">
<!--
@import url(css/estilos.css);
-->
</style>
```

## Cómo crearlos

1. **Redefiniendo etiquetas:** nombramos al estilo igual que la etiqueta que queremos definir. Ejemplo:

```
h1 {} → <h1>texto</h1>
```

2. **Clases:** creamos estilos con el nombre que queramos y con un punto delante para aplicar a las etiquetas mediante el atributo **class=""**. Ejemplo:

```
.estilo {} → <h1 class="estilo">texto</h1>
```

3. **Identificadores:** creamos estilos con el nombre que queramos y con una almohadilla delante para aplicar a las etiquetas mediante el atributo **id=""**. Ejemplo:

```
#estilo {} → <h1 id="estilo">texto</h1>
```

4. **Condicionales:** condicionamos la aplicación de un estilo cuando esté dentro de otro separando las etiquetas/clases/identificadores por espacios, o combinando etiqueta y clase/identificador. Ejemplo:

```
h1 span {} → <h1><span>T</span>exto</h1>
h1.estilo {} → <h1 class="estilo">texto</h1>
```

5. **Selectores:** condicionamos la aplicación de un estilo a un estado de la etiqueta. Ejemplo:

```
a:link, a:active, a:visited, a:hover {}
```

**NOTA:** si separamos diferentes tipos de estilos por comas, se aplicará la misma definición a todos.